MULTIMEDIA UNIVERSITY®

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 2, 2016/2017

ECE1026 ALGORITHMS AND DATA STRUCTURES

(All Sections / Groups)

03 MAR 2017
3:00 P.M – 5:00 P.M.
(2 Hours)

INSTRUCTIONS TO STUDENTS

1. This question paper consists of 8 pages including the cover page with 4 questions only.

2. Attempt all four questions. All questions carry equal marks and the distribution of marks is given in the questions.

3. Please print all your answers in the answer booklet provided.

4. State all assumptions clearly.

Question 1

(a) Draw a finite state diagram for regular expression ((AB)*) | C .

[6 marks]

(b) Refer to Fig. Q1-1 for the following questions:

(i) State the set of all states (Q), the set of all input symbols ($\sum$), the initial state (q_0), and the set of final states (F), for the finite state machine.

[6 marks]

(ii) Find the regular expression of the finite state machine.

[4 marks]

(iii) Give 4 accepted words for the finite state machine.

[4 marks]

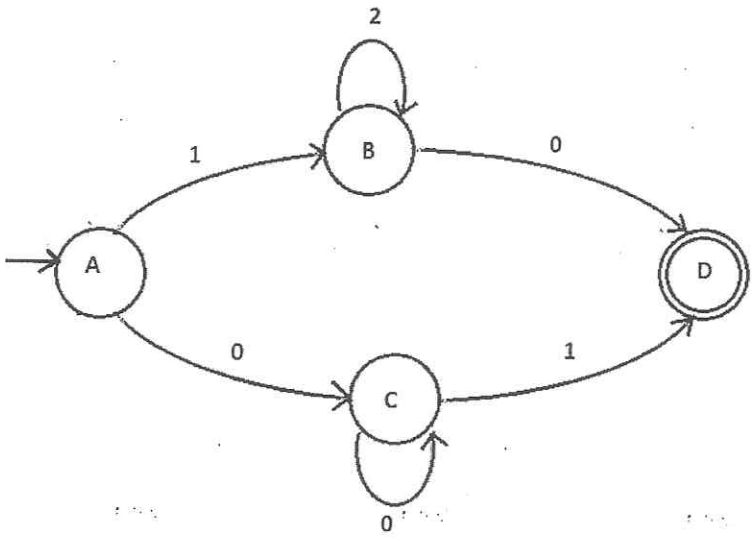

Fig. Q1-1

Continued ...

Question 1 (continued)

(c) Consider the Turing machine in Fig. Q1-2. Write down the tape contents and the read/write head position at the end of **each execution cycle**, until the machine halts.

[5 marks]

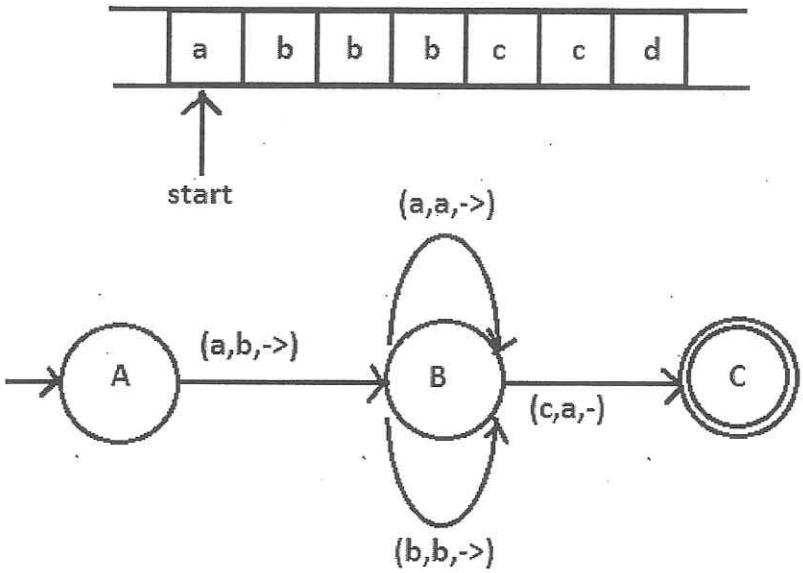

Fig. Q1-2

Continued ...

Question 2

(a) You are required to create a database which records student name, weight and height using linked list. A partial C program is shown in Fig. Q2-1.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

//Answer for part (a)(i) - struct Student definition

struct Student *addStudent(char name[],float weight, float height);

int main(){
    float sum_weight;
    int counter;
    struct Student *head=NULL, *current=NULL;
    head = current = addStudent ("",0, 0.0);

   //Answer for part (a)(iii) - store data

   //Answer for part (a)(iv) - calculate and display average

    return(0);
}

//Answer for part (a)(ii) - function addStudent definition
```

Fig. Q2-1

(i) Write C code to define a suitable linked list node structure **`Student`**.

[3 marks]

(ii) Write a function definition for **`addStudent`** which creates and initializes a new **`Student`** node.

[4 marks]

(iii) Write C code to store data from Table Q2 in a linked list.

[3 marks]

Table Q2

Name	Weight (kg)	Height (cm)
Adam	55.2	155.2
Maryam	45.7	140.5
Jane	58.5	150.8

(iv) Write C code to **calculate** and **display** the **average** weight from data stored in a linked list.

[5 marks]

Continued ...

Question 2 (continued)

(b) Study the following C program in Fig. Q2-2.

```
#include <stdio.h>

void printSet(char name[], int s) {
    int j;
    printf("%s = { ",name);
    for(j=0;j<16; j++)
        if((1<<j)&s) printf("%d ",j);
    printf("}\n");
}

int main() {
    unsigned int setA = (1<<0)|(1<<2)|(1<<7); //part (i)
    unsigned int setB = (1<<1)|(1<<2)|(1<<7)|(1<<9); //part (i)

    //Answer for part (ii)

    printSet("Set A", setA); //part (iii)
    printSet("Set B", setB); //part (iii)

    //Answer for part (iv)

    return 0;
}
```

Fig. Q2-2

(i) Compute the initial decimal values of **setA** and **setB**.

[3 marks]

(ii) Write C code to insert element 12 into **setB**.

[2 marks]

(iii) Determine the output of the first two **printSet** statements in the **main** function (**setB** includes element 12).

[2 marks]

(iv) Write C code to display the union and the intersection of **setA** and **setB**.

[3 marks]

Continued ...

Question 3

(a) Explain direct and mutual recursions. Give an example of mutual recursion in C programming language.

[6 marks]

(b) Fig. Q3-1 shows C code of a recursive program. The program prints "`This is a recursive function`" repeatedly until a point it prints "`Segmentation fault`" and stops. Discuss what happened to the program and why it happened.

[5 marks]

```
#include <stdio.h>

void func(void) {
    printf("This is a recursive function \n");
    func();
}

int main(void) {
    func();
    return 0;
}
```

Fig. Q3-1

(c) The highest common factor of two positive integers can be found using Euclid's algorithm. An iterative implementation of Euclid's algorithm is shown in Fig. Q3-2.

(i) Write down the recursive equations of Euclid's algorithm. Extract them from Fig. Q3-2.

[4 marks]

(ii) Rewrite function **`hcf`** as a recursive function.

[6 marks]

(iii) Write down one advantage and one disadvantage of recursion.

[4 marks]

Continued ...

Question 3 (continued)

```
unsigned int hcf(unsigned int a, unsigned int b) {
   if (a==0) return b; //hcf(0,b) = b
   if (b==0) return a; //hcf(a,0) = a

   while (a != b) {
      if (a > b)
         a = a - b; //repeat with a = a - b
      else
         b = b - a; //repeat with b = b - a
   }
   return a; //hcf found when a == b
}
```

Fig. Q3-2

Continued ...

Question 4

(a) Write an ANSI C function `binSearch` to search a sorted array of integers using a binary search algorithm. The function takes in the integer wanted, an array of integers and some other arguments as necessary e.g. array size or start/end array index. If a match is found, the function returns the index of the matching element, otherwise it returns -1.

[8 marks]

(b) Write an ANSI C function `insertSort` to sort an array of integers in **ascending** order using an insertion sort algorithm. The function takes in an array of integers and the array size. The input array is to be replaced by the sorted array.

[8 marks]

(c) Sort by hand the word "`DISCOVER`" in **ascending** order using the following sorting algorithms. Show all steps involved.

(i) Selection sort

[3 marks]

(ii) Insertion sort

[3 marks]

(iii) Quick sort (the middle element as pivot)

[3 marks]

End of Paper